REMARKS

By this amendment, claims 1-16, 18, 20 and 22-34 are pending. Claims 17, 19 and 21 were previously canceled.

The final Office Action mailed June 17, 2005 rejected claims 1-16, 18, 20 and 22-35 as obvious under 35 U.S.C. § 103 based on *Gelman et al.* (US 6,415,329) in view of *Albert et al.* (US 6,742,045).

Applicants respectfully traverse the outstanding rejection on the merits, because in Applicants' view the claimed invention patentably defines over the applied references, as next discussed.

Independent claims 1 and 29 recite "restarting a spoofed connection between the second platform and a host, wherein the second platform serves as a redundant platform for the first platform." Claim 15 recites "a processor coupled to the communication interface and configured to restart a spoofed connection upon detection of a redundancy switch from a primary gateway." Claim 22 recites "means for restarting a spoofed connection upon detection of the redundancy switch condition."

After the careful study of the references and the rejection, it is apparent that the Examiner misunderstands the claimed invention. First, the claimed invention (e.g., claim 1) is directed to "performing **redundancy switching** from a first platform to a second platform." The *Gelman et al.* system does not contemplate any such capability, and in fact, has no technical support for the "terminating, during a predetermined period, the connection based upon the identifying step," and "restarting a spoofed connection between the second platform and a host, wherein the second platform serves as a redundant platform for the first platform." Notably, the Examiner's interpretation of the claimed terminating step is flawed. For a supposed disclosure of this step, the Examiner refers to FIG. 10, col. 10:1-8, and col. 15: 47 – col. 17: 20. Examination of these passages reveal that the only relevance of such passages is the use of the term "terminated" (col. 10: 8). However, Applicants note that use of the term is not in the same context of the claimed terminating step. *Gelman et al.* discloses, in col. 10: 1-8 and col. 15: 47 – col. 17: 20, the following (Emphasis Added):

> To reach the GT application, TCP packets must have their destination addressing information changed. The GT SNAT 78 module performs this function, selectively changing the destination addressing information of incoming packets which are to be transmitted on the wireless link, to that of the gateway application's main socket, **forcing the connection to be terminated by the TCP/IP protocols on the gateway**. (col. 10: 1-8)

FIG. 10 illustrates how each TCP connection that normally connects directly a client to a server is split into three segments. Note that while a server may also initiate a TCP connection, for simplicity it is assumed that the client initiates a TCP connection, and that the server accepts the TCP connection. Note also that when referring to hosts or nodes, "client" and "source" are generally used synonymously, as are "server" and "destination".

The first segment 210 is a TCP connection, the client TCP connection, whose endpoints are the client 220 and a first gateway, or client gateway 222. The third segment 214 is also a TCP connection, the server TCP connection, whose endpoints are the server 226 and a second gateway, or server gateway 224. These segments 210, 214 may be simple links as shown, or they may comprise networks of arbitrary size. The second segment 212 is a WLP session whose endpoints are the client gateway 222 and server gateway 224. The TCP protocol does not run on this segment 212.

Information pertaining to the three segments, how they relate to each other, and how they relate to the original TCP connection is stored on each gateway in the framework of a virtual circuit (VC). A client VC 228 is initialized on the client gateway 222 when the client gateway 222 recognizes that a client 220 is attempting to make a TCP connection to a server 226. Data originally sent in TCP packets from the client 220 to the server 226 is sent on the client VC 228.

A server VC 230 is started on the server gateway 224 when the server gateway 224 is told by the client gateway 222 that a client 220 has attempted to start a TCP connection with the server 226. Data originally sent in TCP packets from the server 226 to the client 220 is sent on the server VC 230.

Topologically, the only way for the client 220 to communicate with the server 226 is through the client and server gateways, 222 and 224 respectively. These gateways 222, 224 are oriented in a manner such that each client can only access one gateway. Prior to any communication between clients and servers, a WLP session is set up between the two gateways 222, 224 that wish to communicate.

FIG. 11 illustrates the network topology 240 used in this example. Packets originate at clients located within one subnet 242, and are destined for a host within a second subnet 252 located across the link 250. Routers 244 ensure that only packets which need to traverse the link 250 between the gateways 246, 248, and packets which are themselves addressed to the gateways 246, 248 themselves are sent to the gateways.

FIG. 12 illustrates the protocol stacks 260, 262, 264, 266 on the client, client gateway, server gateway, and server respectively, and the procedure through which the GT framework is set up. Normally, when a client starts a TCP connection to a server, the client sends a TCP SYN packet. The server acknowledges the client's SYN by sending its own SYN, and the client acknowledges the server's SYN.

In a preferred GT embodiment of the present invention, when the client attempts to start a TCP connection to the server, the SNAT 268 on the client gateway receives the client's SYN packet and notices that it is destined for the server. The SNAT 268 changes the addressing information in the packet so that the packet is forwarded to the client gateway application 270. The client gateway's TCP module 269 acknowledges the client's SYN and sends its own SYN packet back to the client. The client gateway SNAT 268 modifies this SYN packet so that the client thinks it is coming from the server.

Finally, the client acknowledges the client gateway's SYN and a client TCP connection 301 is established. The client gateway SNAT 268 henceforth modifies all packets for this TCP connection 301 so that incoming packets sent by the client are forwarded to the client gateway, and outgoing packets sent from the client gateway to the client appear to have originated at the server.

> The client gateway application 270 receives information from the SNAT 268 as to with which host the client was attempting to start a TCP connection (step 302). The client gateway application 270 initializes a client circuit 303, and informs the server gateway that the client is trying to connect to the server. The server gateway application 272 then initializes a server circuit 306, informs the server gateway SNAT 274 (step 304) that it is starting a server TCP connection 305 with the server, and sends a TCP SYN packet to the server.
>
> The server gateway SNAT 274 modifies the SYN packet so that the packet appears to have come from the client, and henceforth modifies incoming and outgoing packets in a similar manner as the client gateway SNAT 268. Once the server TCP connection 305 has been successfully established, the server gateway application 272 informs the client gateway application 270 of the success, and the client and server communicate with each other through the gateways.
>
> Although the present invention breaks up end-to-end TCP semantics, this has no bearing on flow control. A client or server gateway can use the TCP receive window buffer size to limit the amount of data that is to be sent over the wireless link. Once the receive buffer of a local connection fills up, the local host is not able to send any more data because it sees congestion on the network. The present invention can read data from the TCP receive buffers in a fair manner so that all TCP connections get a fair share of the wireless link. This is important as flow is still controlled fairly on the terrestrial portion of the network. (col. 15: 47 – col. 17: 20)

Based on the above lengthy passages, it is evident that the connection being "terminated on the gateway," is terminated not in the sense that the connection is torn down or disconnected. By contrast, claim 1 recites "**terminating**, during a predetermined period, **the connection** based upon the identifying step; and **restarting a spoofed connection** between the second platform and a host, wherein the second platform serves as a redundant platform for the first platform." These features are fully supported by Applicants' Specification. For example, the Specification on paragraph [105] details the following:

> As a result of the redundancy switch, all of the TCP connections that were being spoofed by the primary gateway 1201 will eventually terminate (because the redundant gateway 1205 has no TCP spoofing state for these connections). This termination of the spoofed TCP connections may take some time to take effect, thereby prolonging the impact on the network as a result of the redundancy switch. Therefore, the redundant gateway 1205, according to an embodiment of the present invention, is capable of performing startup processing, which is designed to expedite restart of the TCP connections, as more fully described below.

Given the Examiner's interpretation of "terminated" as performed by the *Gelman et al.* system, the claimed restarting step is without any technical merit. Perhaps in recognition of this, the Final Office Action, on page 3, acknowledges that "Gelman does not explicitly disclose restarting a spoofed connection between the second platform and a host wherein the second platform serves as a redundant platform for

the first platform." This consistent with the fact that there is no support within *Gelman et al.* for "restarting" any connection, much less in the manner claimed.

Therefore, a *prima facie* of obviousness thus has not been established. To establish *prima facie* obviousness of a claimed invention, all of the claim limitations must be taught or suggested by the prior art. ***In re Royka***, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). All words in a claim must be considered in judging the patentability of that claim against the prior art. ***In re Wilson***, 424 F.2d 1382, 1385, 165 USPQ 494, 496 (CCPA 1970). In this instance, the Examiner has misinterpreted the claimed terminating step and conveniently ignored the claim language "restarting."

The addition of *Albert et al.* does not fill in the gaps of *Gelman et al.* The Examiner applies *Albert et al.* for a teaching of the use of backup service managers for the purpose of providing a fail over scheme. Even assuming the references were properly combined based on some teaching or suggestion in the references, and assuming the modifications proposed in the Office Action were justified by additional teachings or suggestions found in the references, even the combination does not render the claimed invention obvious. Specifically, none the references taken alone, or in combination, teaches or suggests "terminating, during a predetermined period, the connection based upon the identifying step; and restarting a spoofed connection between the second platform and a host, wherein the second platform serves as a redundant platform for the first platform."

Furthermore, in suggesting the modification of *Gelman et al.* by utilizing the teachings of *Albert et al.*, Applicants respectfully submit that the Examiner does so based on hindsight reconstruction of Applicants' claims, as the references themselves lack support for the proposed combination. *Albert et al.* discloses that service manager 270 may also include a service manager interface 282 used to communicate with other service managers. The service manager may communicate with other service managers for the purpose of providing a fail over scheme of backup service managers. At best, this disclosure provides in general terms a need for a backup capability. In stark contrast, the claims specifically recite "**restarting** a spoofed connection **between the second platform and a host**, wherein the second platform serves as a redundant platform for the first platform." The Examiner fails to appreciate the interaction between the first platform and the second platform to manage the connections in support of "redundancy switching." It is well settled that it is impermissible simply to engage in hindsight reconstruction of the claimed invention, using Applicant's structure as a template and selecting elements

from the references to fill in the gaps. ***In re Gorman***, 933 F.2d 982, 18 USPQ2d 1885 (Fed. Cir. 1991). Recognizing, after the fact, that a modification of the prior art would provide an improvement or advantage, without suggestion thereof by the prior art, rather than dictating a conclusion of obviousness, is an indication of improper application of hindsight considerations. Simplicity and hindsight are not proper criteria for resolving obviousness. ***In re Warner***, 397 F.2d 1011, 154 USPQ 173 (CCPA 1967).

In view of the foregoing discussion, Applicants respectfully urge the indication that independent claims 1, 8, 15, 22 and 29 are allowable. Additionally, dependent claims 2-7, 9-14, 16, 18, 20, 23-28 and 30-35 are allowable at least for the reasons put forth for the allowability of their corresponding independent claims. Further, these dependent claims are allowable on their own merits. For example, claim 2 recites "selectively forward TCP segments unspoofed to a remote platform." This is not technically possible within the *Gelman et al.* system, as all TCP connections (according to the Examiner's interpretation of "spoofing") are by definition spoofed in *Gelman et al.* That is, no TCP segment can be forwarded unspoofed. The Examiner refers to col. 9: 16-col. 10: 37 for a supposed disclosure of this feature; however, this passage simply explains the operation of the WLP module, which is consistent with Applicants' contention.

Therefore, the present application, as amended, overcomes the objections and rejections of record and is in condition for allowance. Favorable consideration of this application is respectfully requested. If any unresolved issues remain, it is respectfully requested that the Examiner telephone the undersigned attorney at (301) 601-7252 so that such issues may be resolved as expeditiously as possible. All correspondence should continue to be directed to our below-listed address.

Respectfully submitted,

9-8-05

Craig L. Plastrik
Attorney for Applicant
Registration No. 41,254